



## An industrial case study on SOA quality evaluation

Riad Belkhatir, Mourad Chabane Oussalah, Arnaud Viguiers

### ► To cite this version:

Riad Belkhatir, Mourad Chabane Oussalah, Arnaud Viguiers. An industrial case study on SOA quality evaluation. International Journal on Engineering Applications (IREA), 2013, pp.2281-2881. hal-00991611

**HAL Id: hal-00991611**

**<https://hal.science/hal-00991611>**

Submitted on 10 Jun 2014

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# AN INDUSTRIAL CASE STUDY ON SOA QUALITY EVALUATION

**Riad Belkhatir**  
Department of Computing  
University of Nantes  
Nantes, France

**Riad.Belkhatir@univ-nantes.fr**

**Mourad Oussalah**  
Department of Computing  
University of Nantes  
Nantes, France

**Mourad.Oussalah@univ-nantes.fr**

**Arnaud Viguier**  
Research/Development Department  
BeOtic  
Rezé, France

**Arnaud.Viguier@beotic.com**

## Abstract

During these last years, Service Oriented Architecture (SOA) has known a meteoric rise and more and more companies are lured by this technology and its strengths (reusability, costs benefits and productivity increase) because of an improved control of the business expectations. This technology could bring a lot of benefits but there may also appear some major complications while disrupting the company organization to adopt it. First and foremost among these, is the risk of not being able to answer favourably to expectations in terms of quality of services. As these risks are distributed through all the services, the question of evaluating SOA has recently arisen. In this light, before adopting SOA, it is fundamental to evaluate the quality of the architecture to set up. This paper presents a tool enabling the assessment of a software oriented architecture based on a model called SOAQE allowing architecture decomposition with the aim of evaluating it easier. The SOAQE model, validated by the software engineering community, served as a basis for the elaboration of this new generation of tools returning results under textual and graphical forms for a better understanding of data.

## Keywords

Software architecture paradigms. Service oriented architecture. Quality attributes.

## 1. INTRODUCTION

Recently, more and more companies focus on SOA solutions for developing their architecture. However, because of the complex nature of the financial issues that this technology involves, there exists a real need in assessing the coherence of the project and the quality of the architecture chosen. This would essentially allow:

- (i) Controlling different costs.
- (ii) Bringing much more credibility to the project.
- (iii) Distinguishing itself from the competition.
- (iv) Leading to certifications (standards).
- (v) Preventing any future significant potential threat including project failures that such evolution could potentially lead to.

Moreover, increases in terms of software size make the development more complex to handle, and this same complexity makes any form of predictability or estimation (cost and quality) extremely difficult. There exists a need to first build a predictive model of quality. We propose in this article a new semi-automated method for evaluating SOAs, called SOAQE (for Service Oriented Architecture Quality Evaluation). This method considerably overcame shortcomings observed so far such as lacks of pertinence and accuracy. The McCall model, which describes software quality and led to the international standard for the evaluation of software quality, the ISO/IEC 9126-1:2001 [1] (which has recently been updated to the SQuaRE standard ISO/IEC 25010:2011 [2]) serves as a basis for our work. Correlatively, we work with a model that can be defined by a set of views and each view is divided in several factors, criteria and metrics. Our experimentations led us to implement a tool called the SOAQE tool (Flex Client/Java Server application), which, based on the SOAQE model, allows quantifying numerically the quality of the architectural point of view branch and all the attributes of its structure. We deal with some state of the art works in the next section then we present the case study from the BeOtic company in Section 3. Section 4 introduces the SOAQE tool which supports our model and Section 5 is devoted to the discussion. Finally, section 6 concludes this paper.

## 2. STATE OF THE ARTS WORKS

The software engineering community first developed methods such as GQM (Goal/Question/Metrics) [3] consisting in a few steps:

1. Define goal of measurement
2. Devise suitable set of questions
3. Associate metric with every question.

The limits of such methods appeared quickly: the fact that the process cannot be automated because the different goals of measurement and the questions/metrics resulting from these goals are exclusively set by stakeholders (human intervention) distorts results because stakeholders are not able to cover all the possible requirements to evaluate the quality.

We have then seen emerge very similar methods like ATAM or SAAM [4] which propelled software architecture evaluation to a standard stage for any paradigm. However, several major concerns have been raised with these methods [4]; in particular their cost in terms of time (a lot of steps to perform the whole process) and money because of the hand operated nature of the evaluations conducted. And again, the major lack concerned the results of the evaluations supported with these methods: lots of deficiencies concerning the requirements of the architecture because the process is still not automated.

The scale of the task has brought the academic world to tackle these issues and to try to develop a more formal and generic approach than different existing methods to evaluate SOAs [4].

New efforts to evaluate SOA are being undertaken in different aspects using different tools and methods like [5] in which they applied attack graphs for SOA security metrics. But the majority of these kinds of researches are just a proposal or they are about some certain aspects of evaluation or using different techniques [6]

From a global perspective, current methods of evaluation are too vague when it comes to giving accurate measures to quality. Our work differs from those existing insofar as we wish to obtain a precise quantitative measurement for each quality factor with our model.

### 3. CASE STUDY

This section describes an extract of a case study of an existing BeOtic's project (<http://www.beotic.com/>).

This case study has not a purpose of validating our method that we already explained in details in a past paper [7] but illustrating it.

#### 3.1 Requirements

For our case study, we collected data from an existing project of the BeOtic Company. These confidential data include code from the service oriented architecture of one of the clients of the company. More exactly, the company implemented its own tool called BeoMetric for collecting metrics from the code (LOC, CR, CCN...); functioning as JMetric (<http://sourceforge.net/projects/jmetric>) and we had the chance to gather XML files regrouping the values of the metrics considered for each method, class and package of the client project.

#### 3.2 Method Use

One of our past works [7] is dedicated to the realisation of the SOAQE model. In [7], we consider that the architectural point of view of an SOA is composed by three main factors (dynamism, reusability and composability) affected by different coefficients according to their importance for SOA (see figure 1).

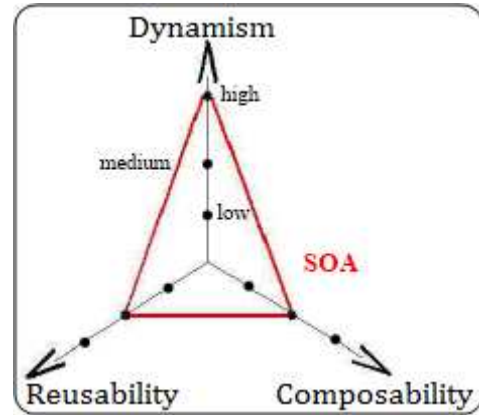


Figure 1: SOA interest points

And each of these factors is composed by the same six criteria (Loose coupling, upgradability, communication abstraction, owner's responsibility, explicit architecture and expressive power) to which we allocate a different weight according to the factor considered (see figure 2).

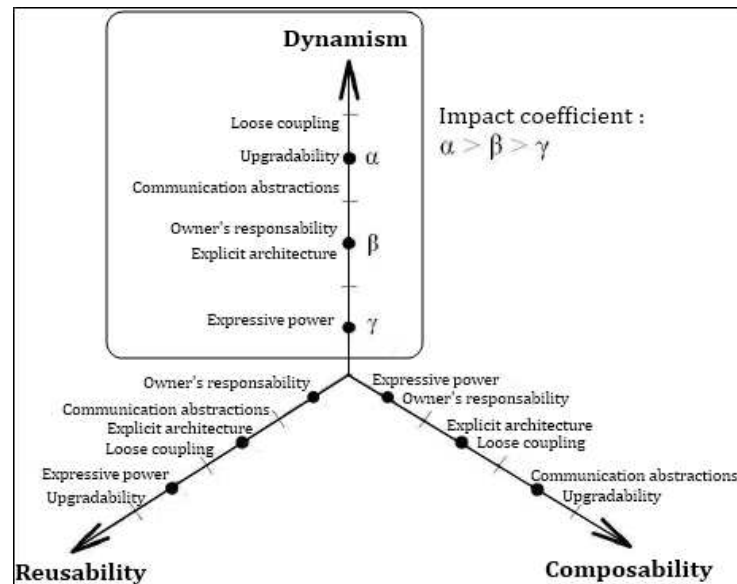


Figure 2: Expression of reusability, composability and dynamism perspectives.

Our first work prompted us to study closely the loose coupling criterion for which we defined its constituent metrics.

The aggregation of the values of these metrics allows obtaining a finite value for the loose coupling criterion (see figure 3).

Therefore, we wish to incorporate to the SOAQE model, the metrics obtained after applying the BeoMetric module to the submitted architecture in order to get a final mark for the quality of the architecture.

The current state of our research works allows us to work exclusively on the path indicated with a blue circle on figure 3 (the loose coupling criterion).

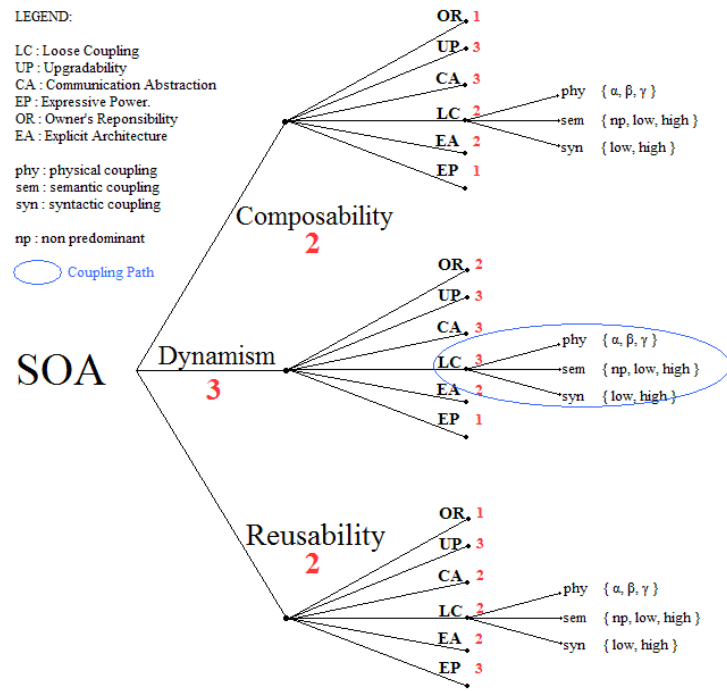


Figure 3: SOA attributes tree weighted with means of coefficients

#### 4. THE SOAQE TOOL

In this section, we present SOAQE Tool (Service Oriented Architecture Quality Evaluation Tool), a tool that supports our method.

##### 4.1 Technical architecture

This prototype has been built in cooperation with the BeOtic company to be used as a service for its customers. The application takes as input XML files where are stocked the values of twenty-six metrics for each method, class and package of the architecture submitted. All these values are then stocked in a SQL database to facilitate data retrieving for the application. The server has been built using Java and the server and the database communicate together via the DAO technology. The client of the application has been implemented using Flex and communicates with the server using Blaze DS. Figure 4 describes the architecture of the SOAQE tool.

##### 4.2 General Organization

The first step of the application consists in displaying in a data grid the set of metric values retrieved from the SQL database. According to the user's choice, these values can be displayed for the classes or the packages of the source code. This is to allow the user to compare the metrics desired for the evaluation before launching it. As show in figure 5, we implemented for the

application a cube stack for the visualization of the results and improved ergonomics.

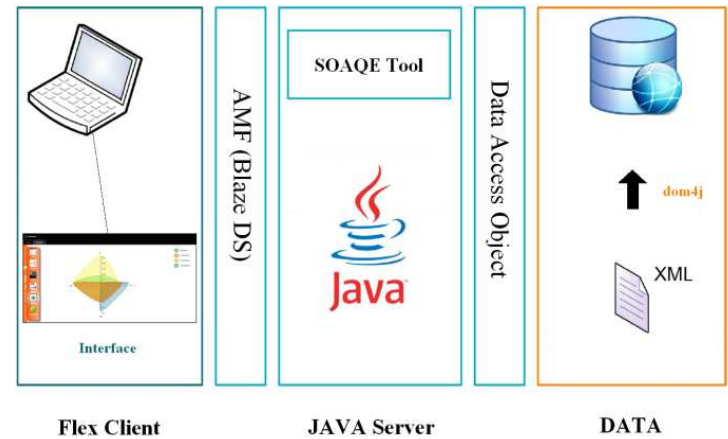


Figure 4 : Architecture of the SOAQE tool.



Figure 5 : Graphical visualisation of the metric values

In this light, the user can see, in addition to the data grid, the behavior of the metric values with the help of a scatter plot composed by three axes corresponding to the classes or packages that the user chooses for the comparison. We also implemented another module where, this time, the user can visualize the evolution of the metric values for each class in the architecture through colored curves (see figure 6).

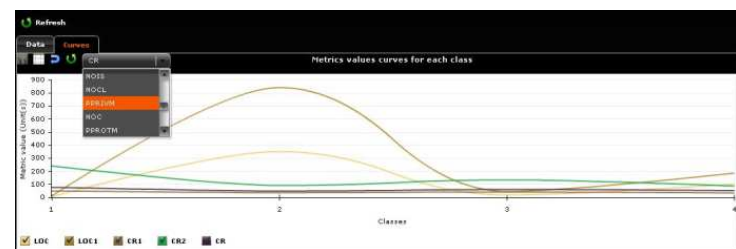


Figure 6: Curves module of the SOAQE tool.

Before launching the evaluation of the architecture submitted, the user can set the tree view of the part of the architecture being evaluated (organised under points of view, factors, criteria and the metrics which has been displayed from the database in the previous phase). The structure of the arborescence is set with a panel under the form of a data grid where is first displayed a default

tree corresponding to the most complete declination of the architecture for the architectural point of view we concluded in a past work [7].

Nevertheless, we offered the possibility to the user to be totally free with his evaluation; this is why it is still possible:

- (i) To modify the attributes selected in the default arborescence.
- (ii) To add new attributes.
- (iii) To delete existing attributes

It has been concluded in past works that only factors and criteria must have corresponding weights because the latter have not the same importance according to the point of view considered. The figure 7 is an overview of this control panel.

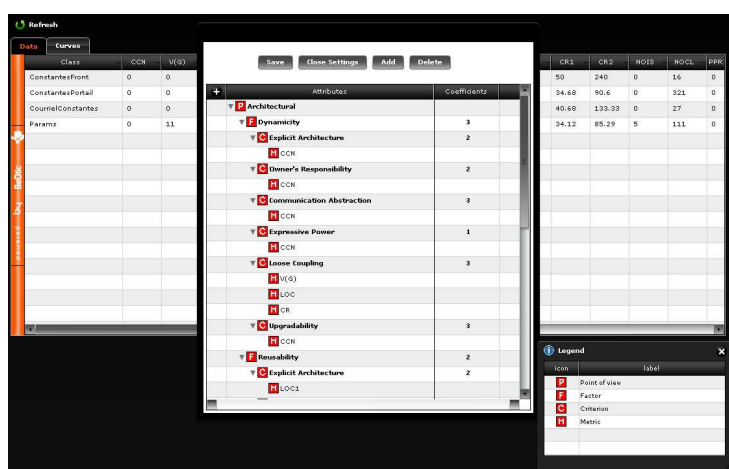


Figure 7: Control Panel

By clicking on the save button, the new arborescence the user created is directly stocked in the SQL database for the next step of the application: the evaluation. Correlatively, the panel closes and a new “Launching the evaluation” button appears. This new operation consists in obtaining a finite value for the quality of the architecture submitted.

(Because the graphics rendering of the results is not only textual, the BeOtic company asked us to not disclose any overview of the graphics rendering to avoid any potential leaks.)

## 5. DISCUSSION

Our proposition offers a new way of evaluating the quality of a service oriented architecture since the process is semi-automated and allows save time and money contrary to all existing works trying to evaluate the quality of an SOA [3, 4, 5, 6]. The model in which the tool is based has always been validated by the software engineering community [7] and allows obtaining real, accurate and immediate results for the

quality evaluation of the SOA. This tool has been implemented to avoid major project fails. Indeed, we can now know if it makes sens to swing towards SOA technology for the company involved and this is exactly where the BeOtic company has an interest in the project because the company is specialised in IT auditing and software distribution.

Nevertheless, we worked on this project as architects and the work for the architectural point of view is not finished as there still are criteria which have not been decomposed in aggregations of metrics. So even if the tool works well and the results obtained are correct, it is still possible to bring new elements to the current work. This is why we chose to let the user free to modify the default arborescence proposed for new research results which are going to be revealed with future works. We first designed a work rather restricted but when the prototype considerably evolved, we added new functionalities to have the most configurable tool possible for the user.

## 6. CONCLUSION

In this paper, we present a model, the SOAQE model that allows splitting and evaluating the quality of a service oriented architecture. The method is based on two main steps:

- (i) The division of the architecture into four levels of attributes (points of view, factors, criteria and metrics).
- (ii) The calculation of the quality mark.

The SOAQE tool has been implemented according to the SOAQE model [7] in order to allow evaluating any SOA considered according to our method.

Further step concerns the deep study of new criteria for the architectural point of view. Correlatively, to obtain a model and a tool which can evaluate in a complete way the quality of any SOA, it is essential to be able to split the whole architecture in a combination of several attributes.

Another part of the perspectives concerns research on new points of views, we already started a bit with the business one.

## 7. REFERENCES

- [1] J.P. Carvalho, X. Franch “Extending the ISO/IEC 9126-1 Quality Model with Non-Technical Factors for COTS Components Selection”
- [2] W. Suryn, A. ABran, A. April, “ISO/IEC SQuaRE: The second generation of standards for software product quality” (2003)

- [3] V.R. Basili, G. Caldiera, H. Dieter Rmbach, "The Goal Question Metric Approach" Chapter in Encyclopedia of Software Engineering, Wiley, 1994
- [4] P. Clements, R. Kazman and M. Klein, Evaluating Software Architectures: Methods and case studied, published
- [5] J. Magott, M.Woda "Evaluation of SOA security metricsusing attack graphs", IEEE 2008, pp 277-284.
- [6] D.Cotroneo, C.Di Flora, S.Russo "Improving Dependability of Service Oriented Architectures for Pervasive Computing", Proceedings of The Eighth IEEE International Workshop on Object-OrientedReal-Time, 2003
- [7] R. Belkhatir, M. Oussalah, and A. Viguiet, A Model Introducing SOAs Quality Attributes Decomposition. ;In Proceedings of SEKE. 2012, 324-327.